

Fig. 1

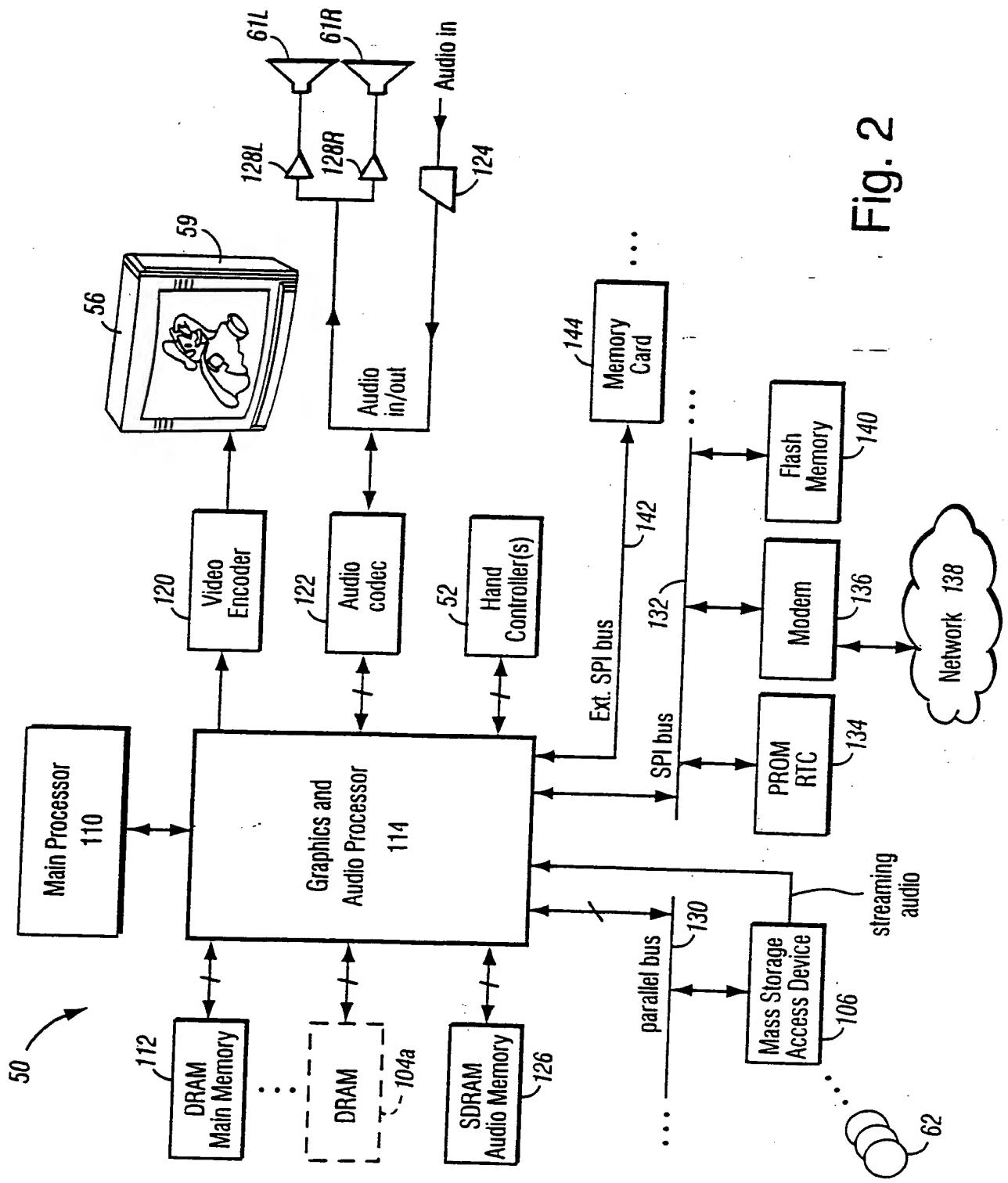
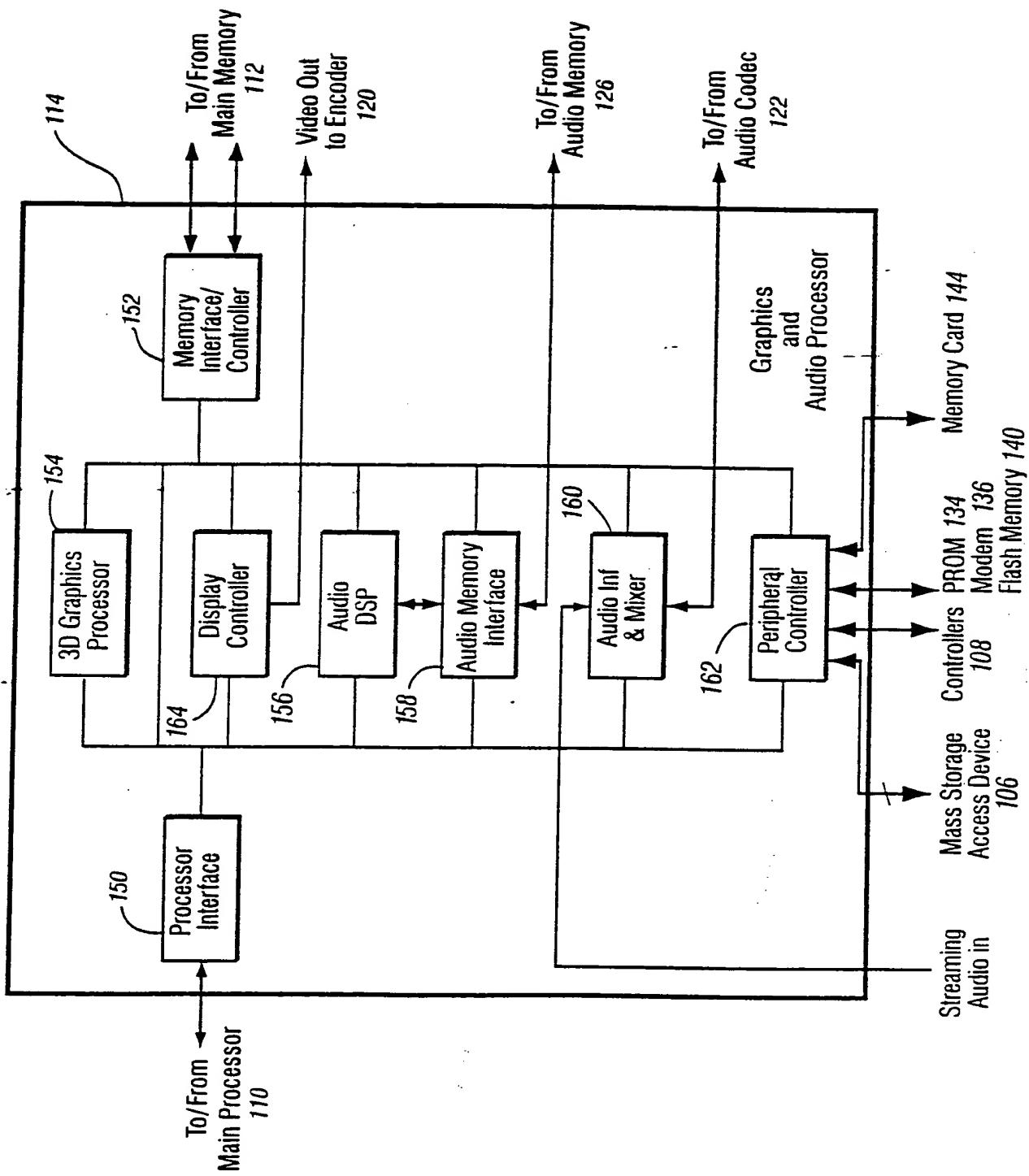


Fig. 2

Fig. 3



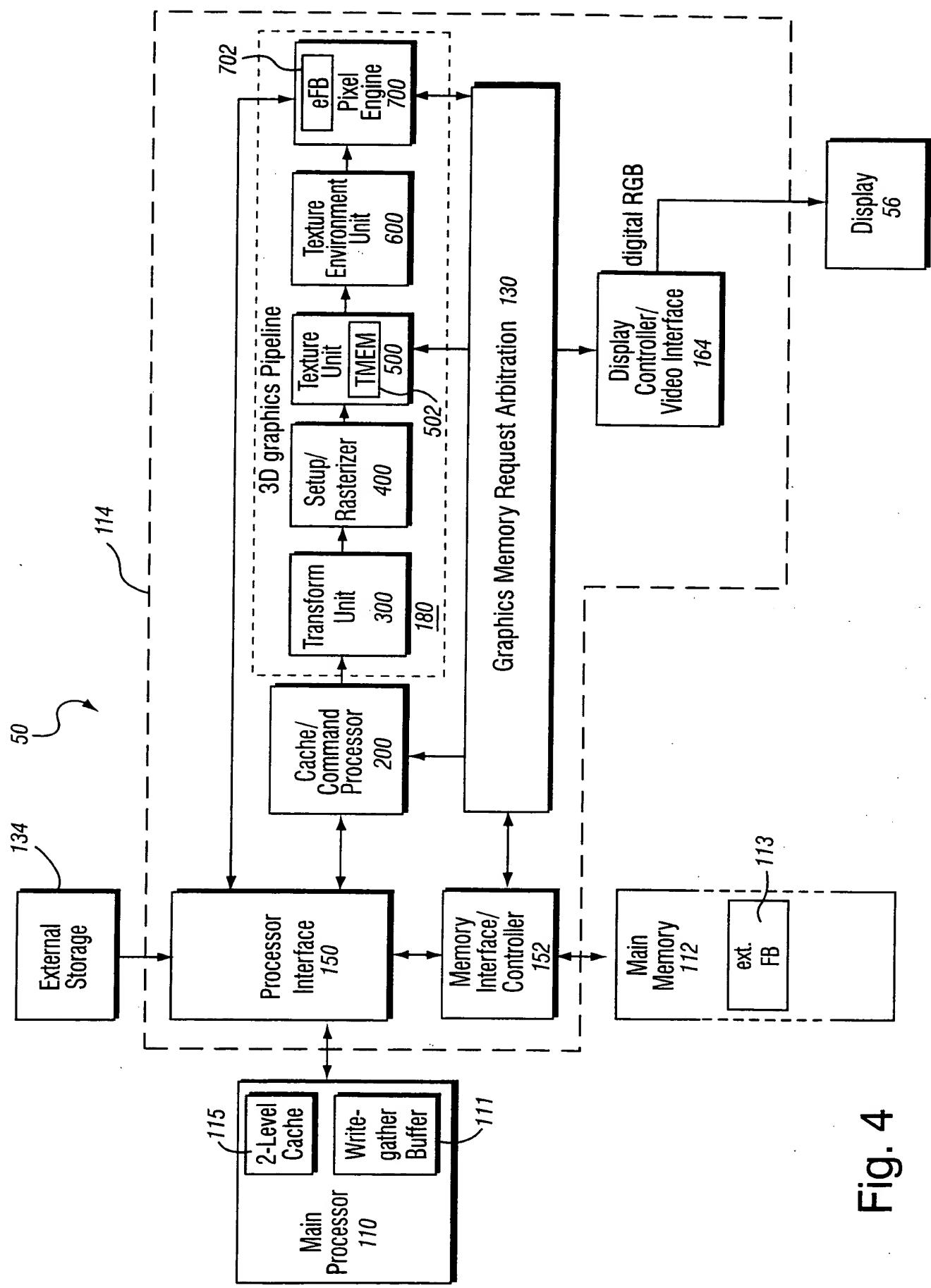


Fig. 4

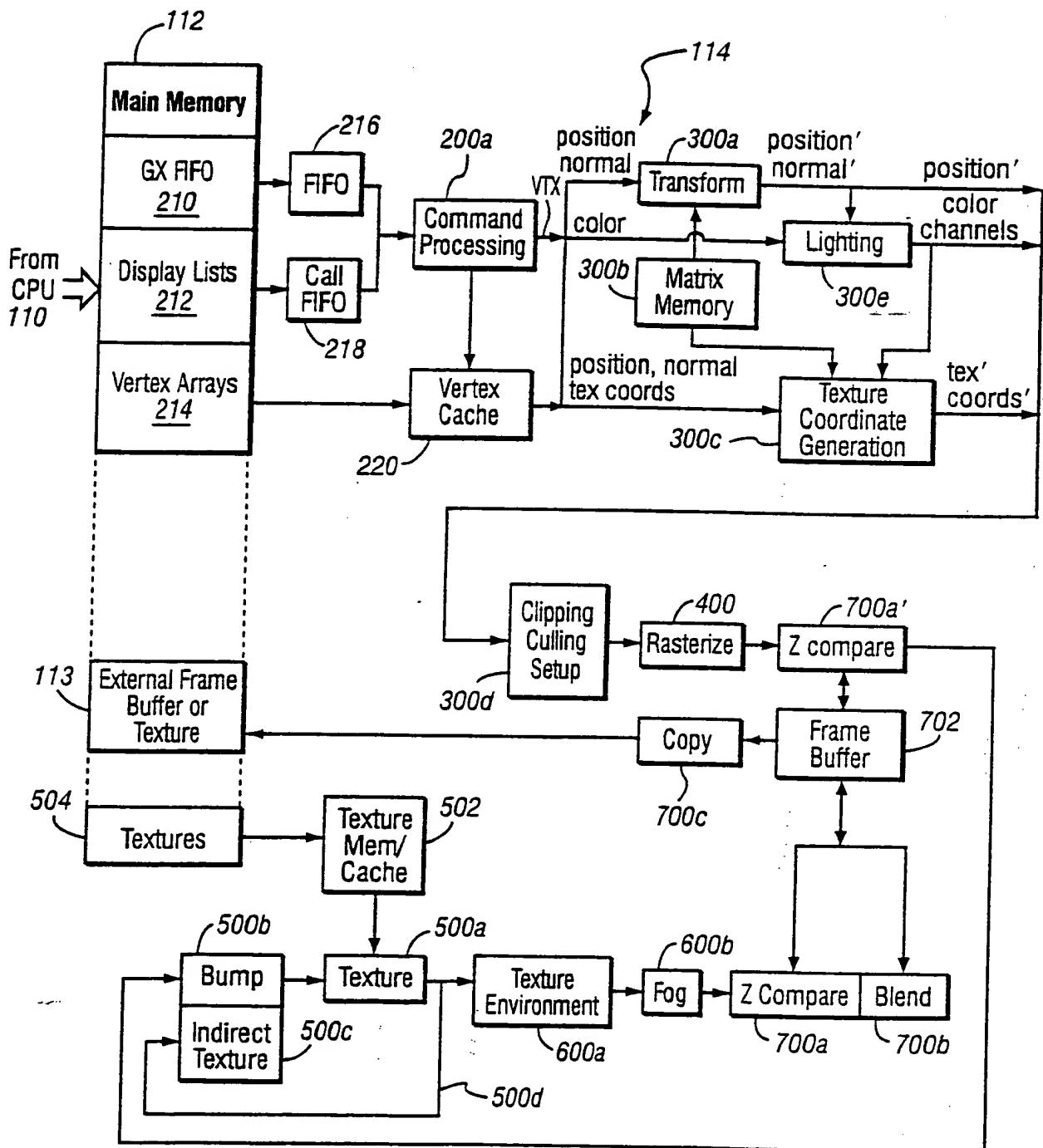


Fig. 5 EXAMPLE GRAPHICS PROCESSOR FLOW

Fig. 6
(Embedded Frame Buffer (EFB))

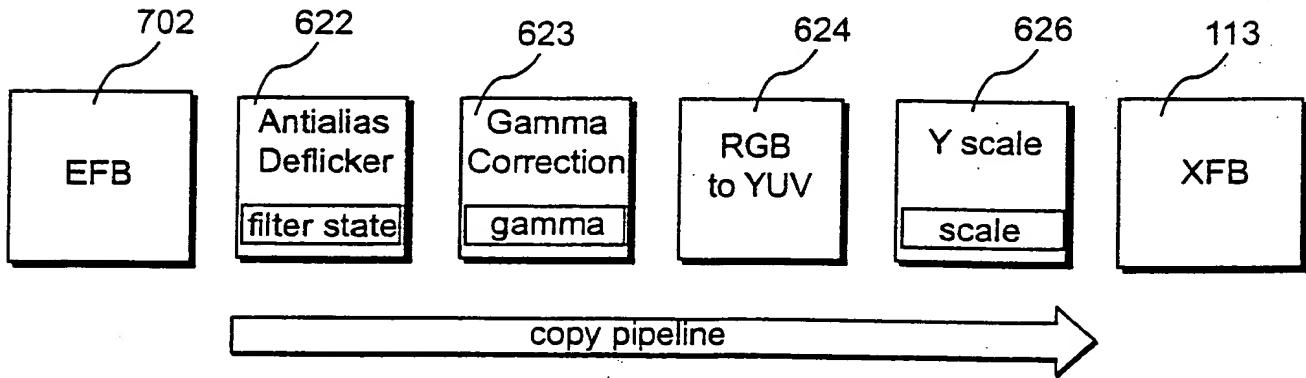
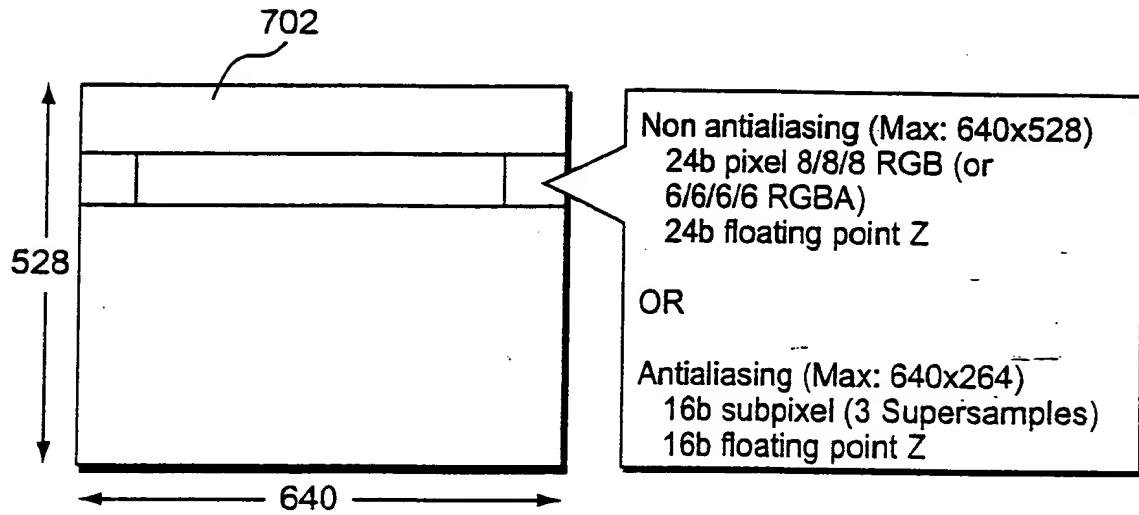
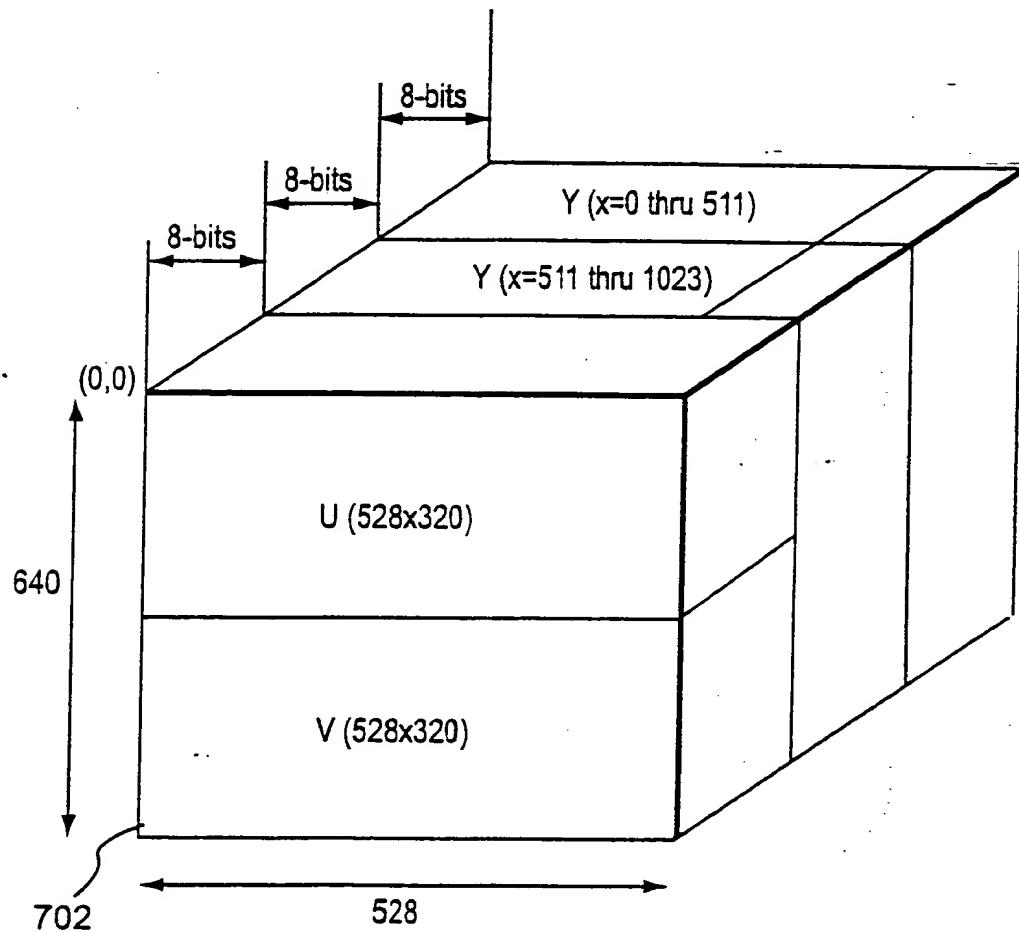


Fig. 9

Fig. 7
(Embedded Frame Buffer Organization)



0007226600 112800

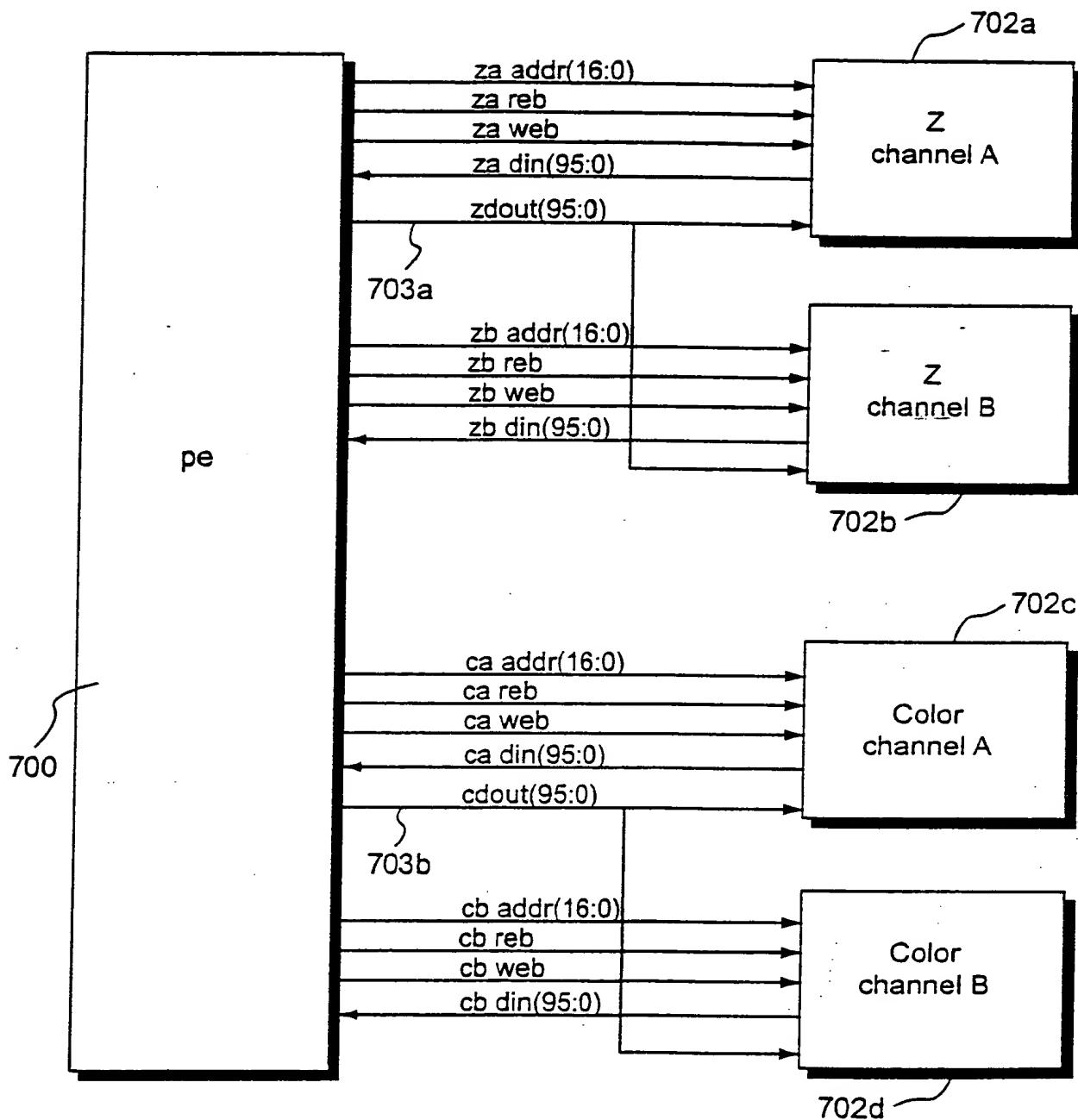
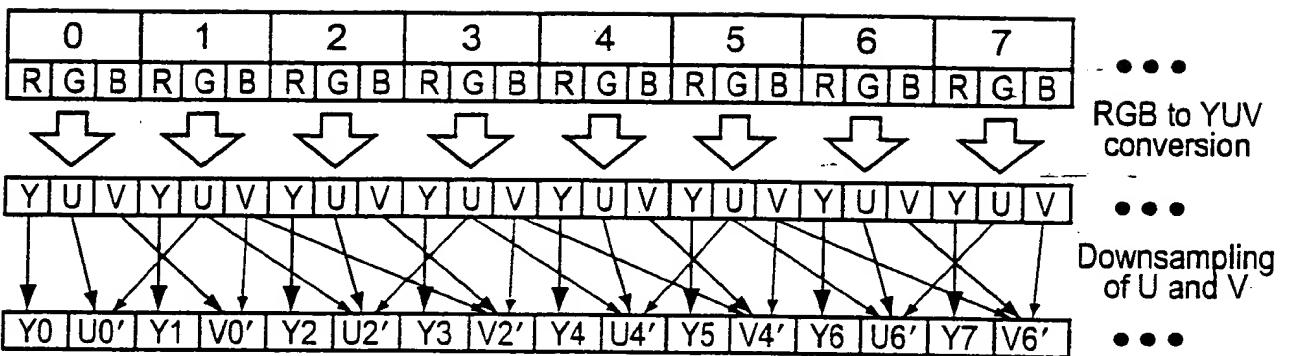


Fig. 8
(Pixel Engine/Frame Buffer Interface)

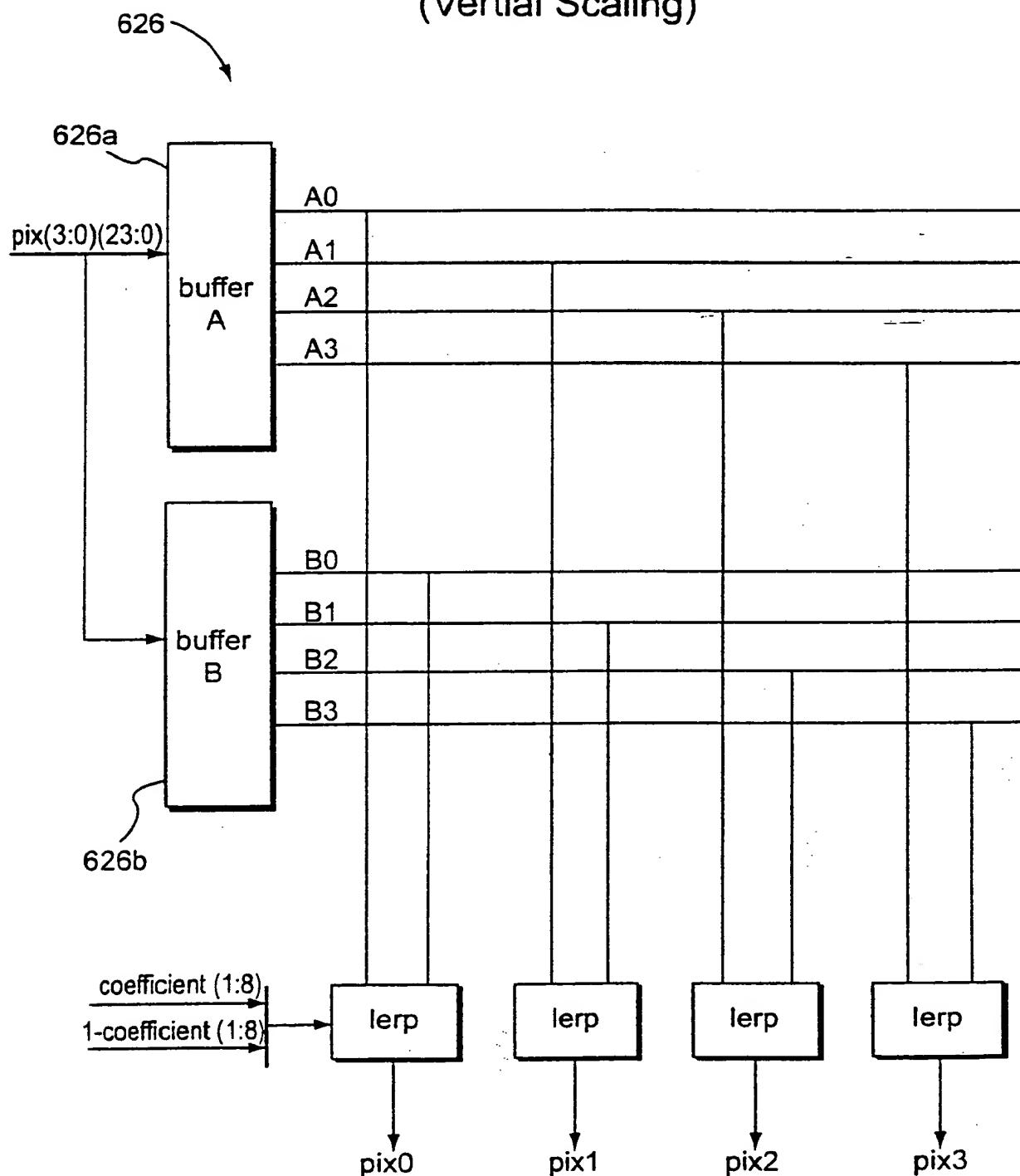


$$U(i) = \frac{1}{4} * U(i-1) + \frac{1}{2} * U(i) + \frac{1}{4} * U(i+1)$$

$$V(i) = \frac{1}{4} * V(i-1) + \frac{1}{2} * V(i) + \frac{1}{4} * V(i+1)$$

Fig. 10A
(RGB to YUV Conversion)

Fig. 10B
(Vertical Scaling)



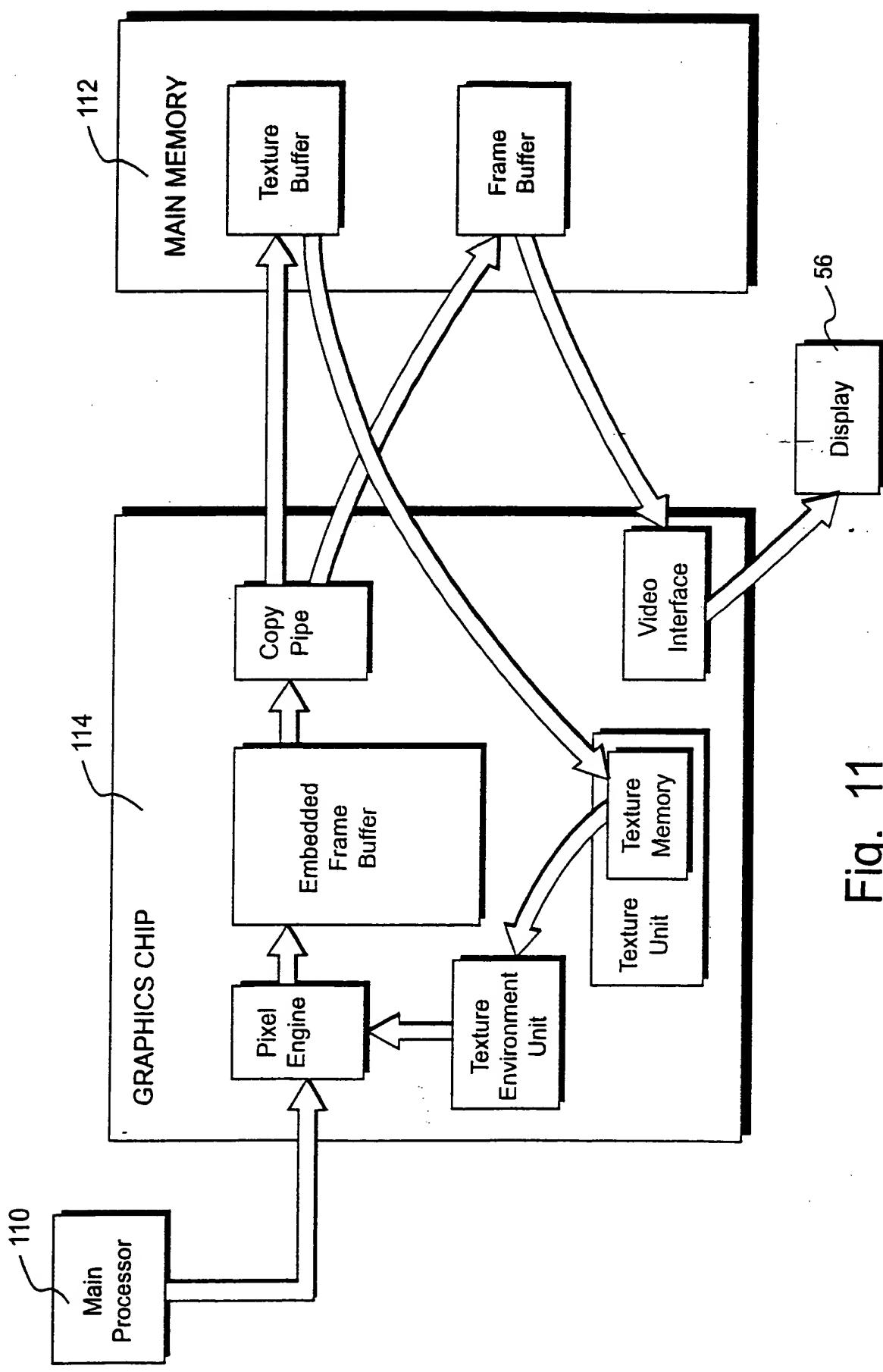
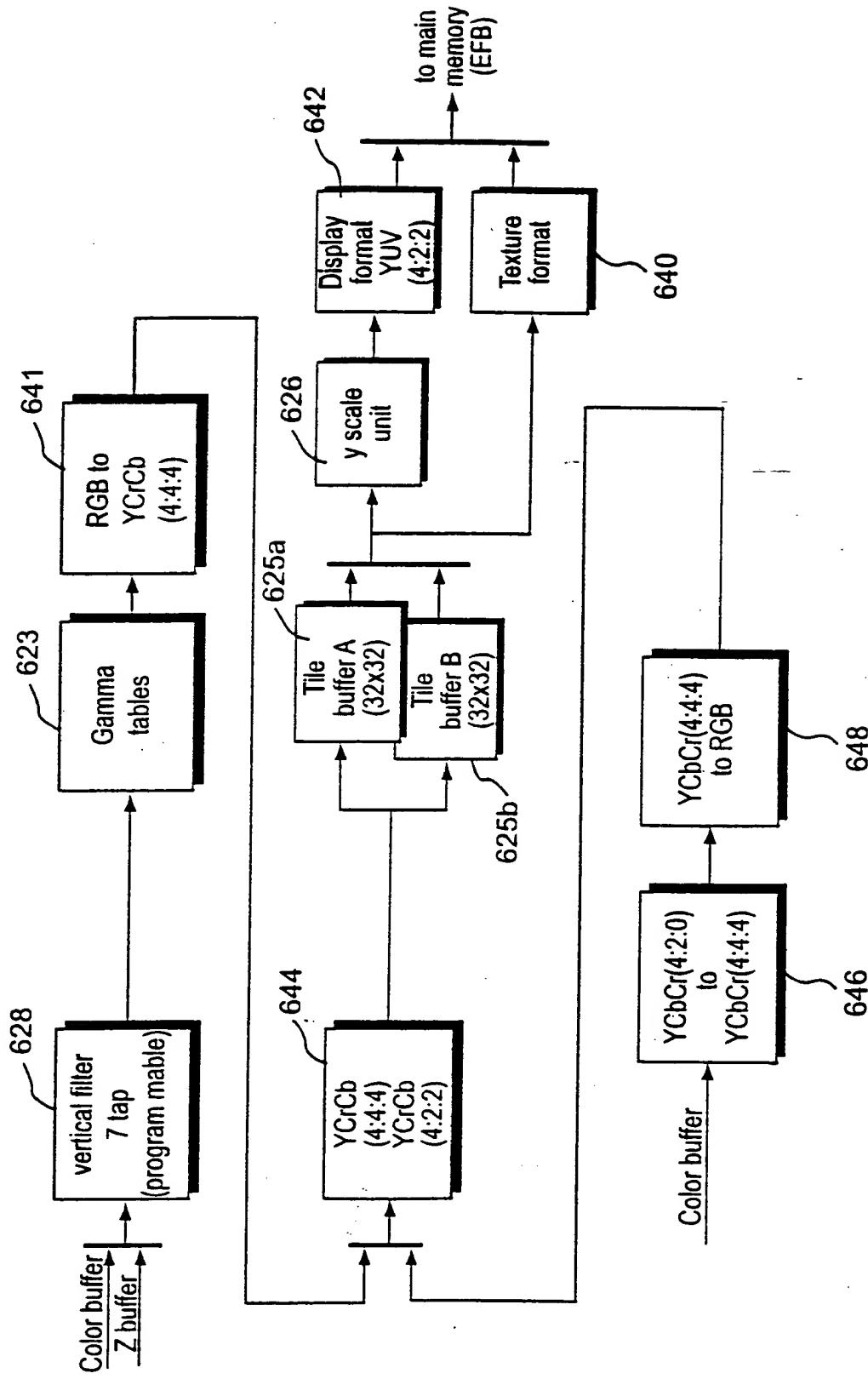


Fig. 11

Fig. 12
(Copy pipeline)



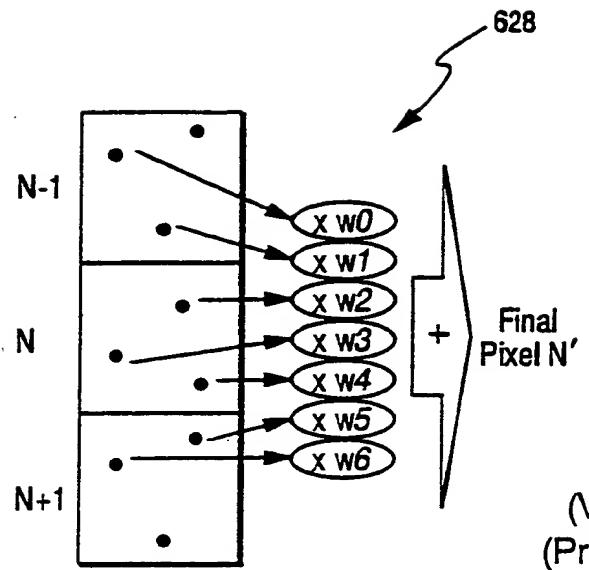


Fig. 12A
(Vertical Filter Blending)
(Programmable 7-tap filter)

0082FF-08E222X60

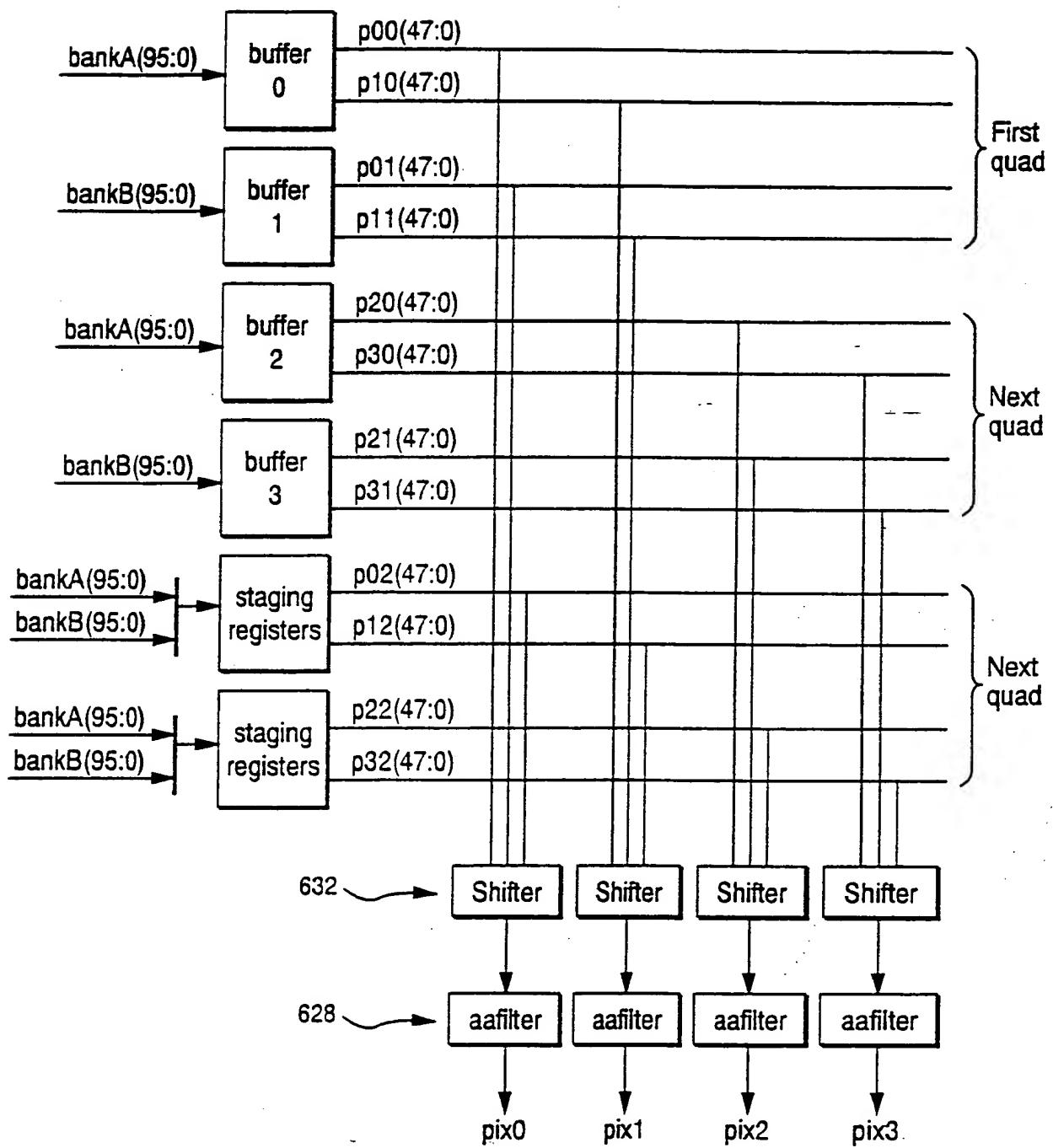


Fig. 12B
(AA buffering)

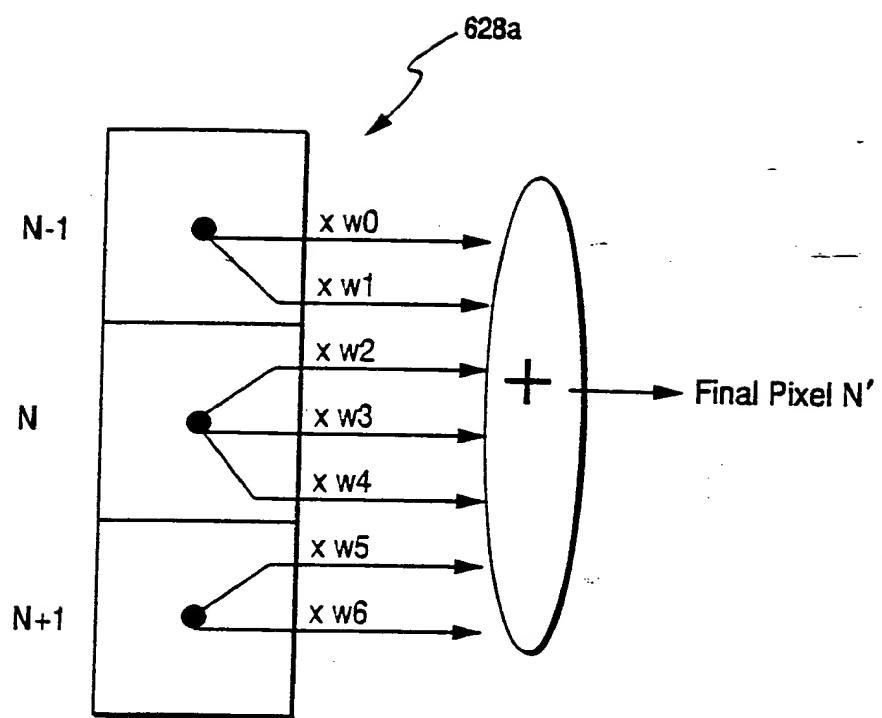


Fig. 12C
Example de-flickering filter

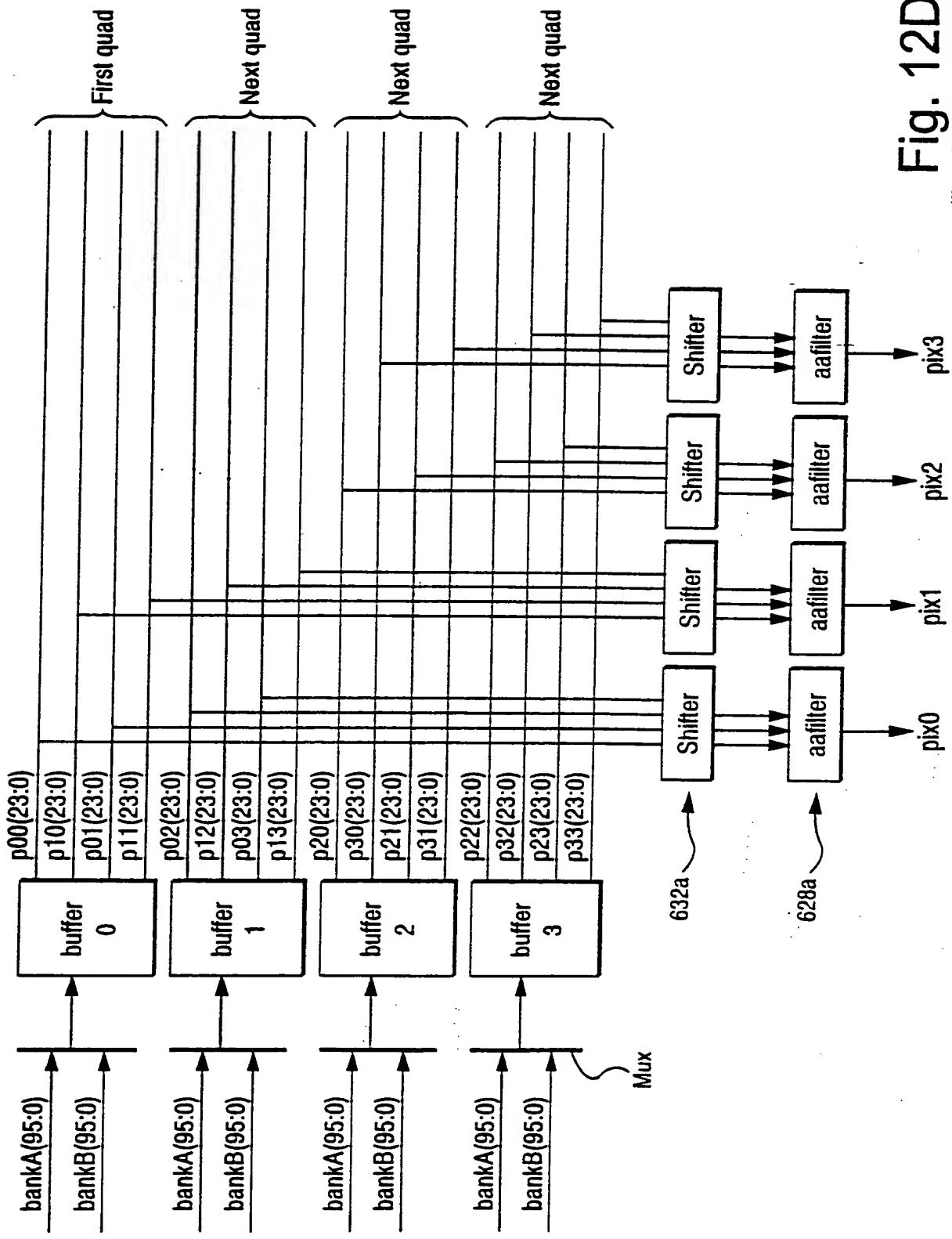


Fig. 12D
(De-flicker buffering)

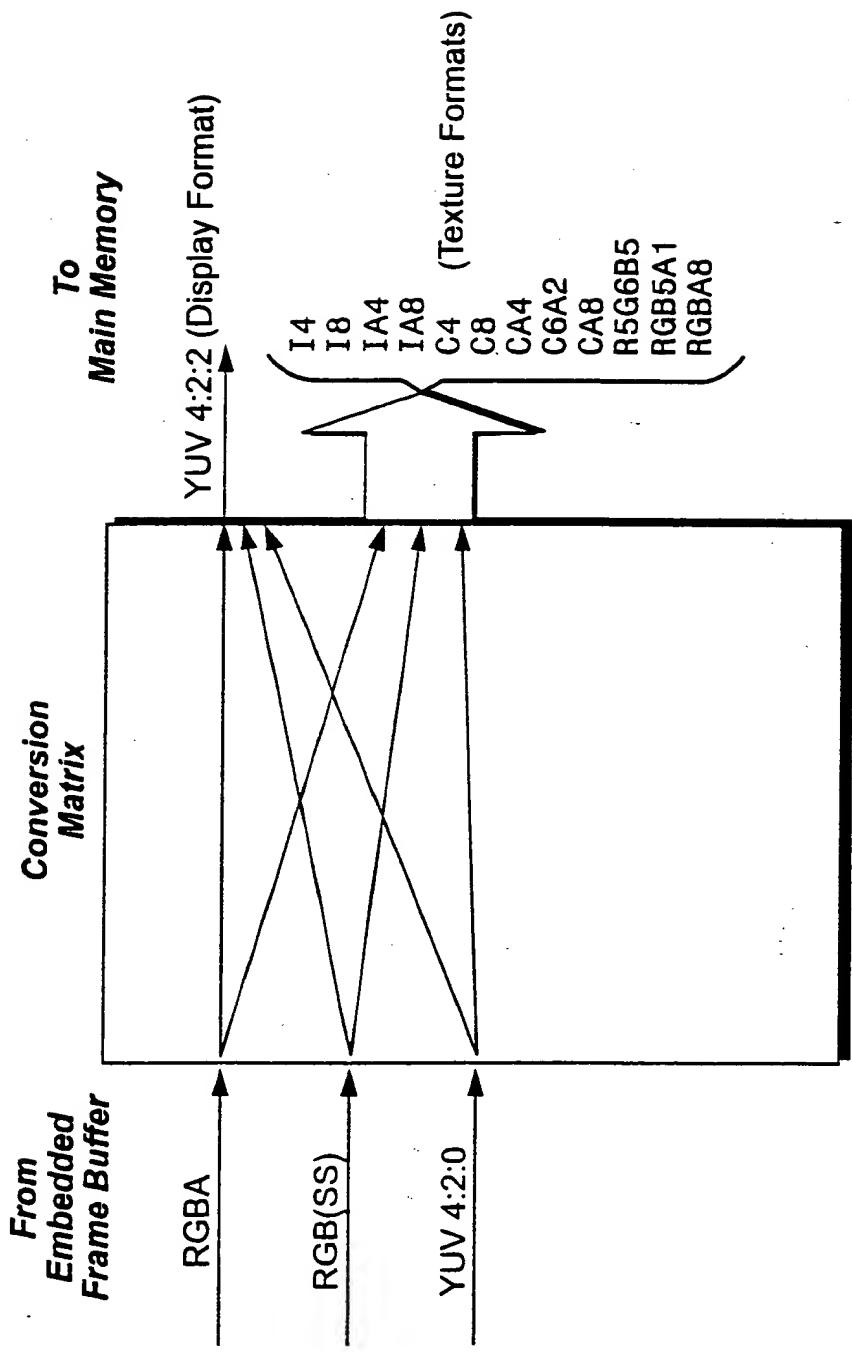


Fig. 12E

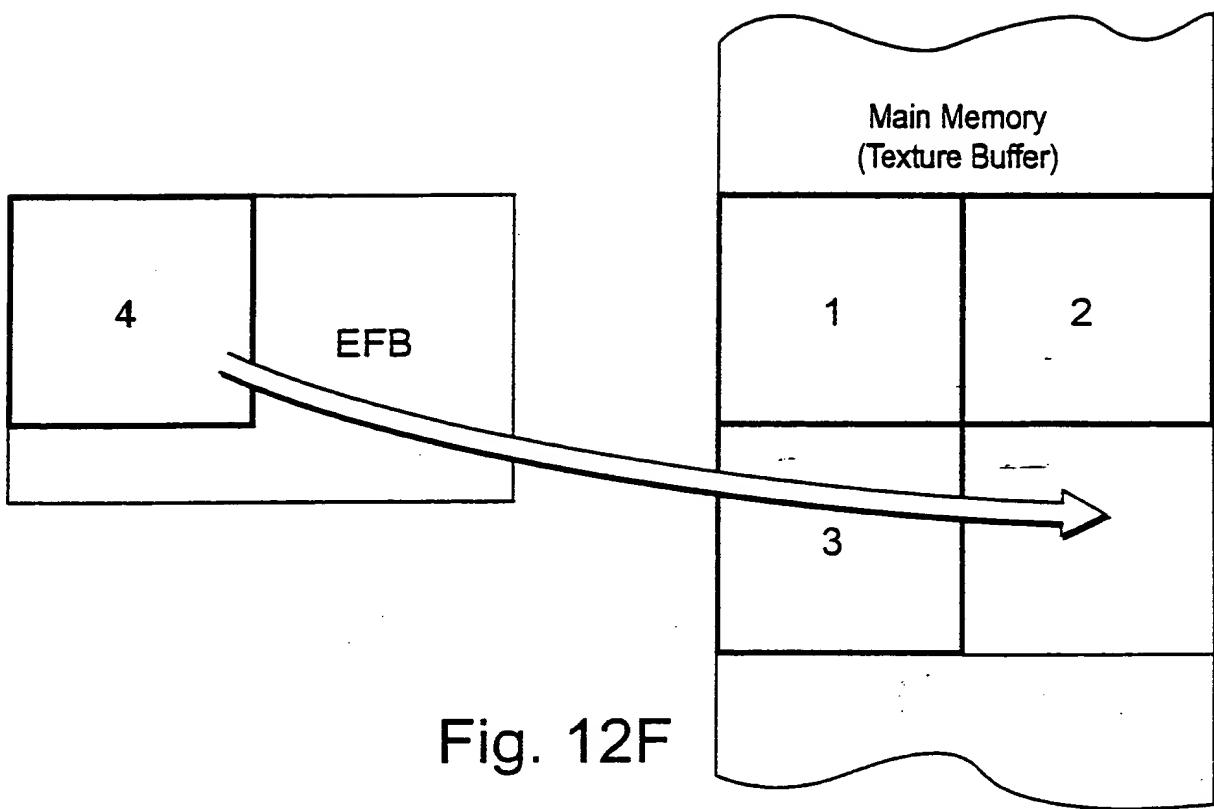


Fig. 12F

Fig. 13
(YCbCr 4:4:4 to 4:2:2 down sampling)

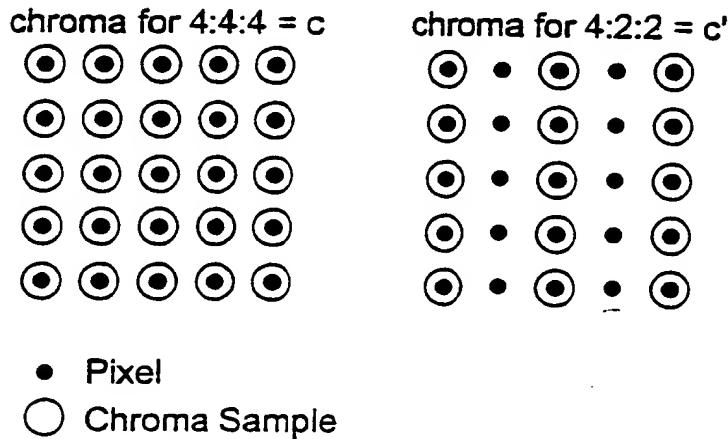


Fig. 14A
(YCbCr 4:2:0 to YCbCr 4:2:2 up-sampling)

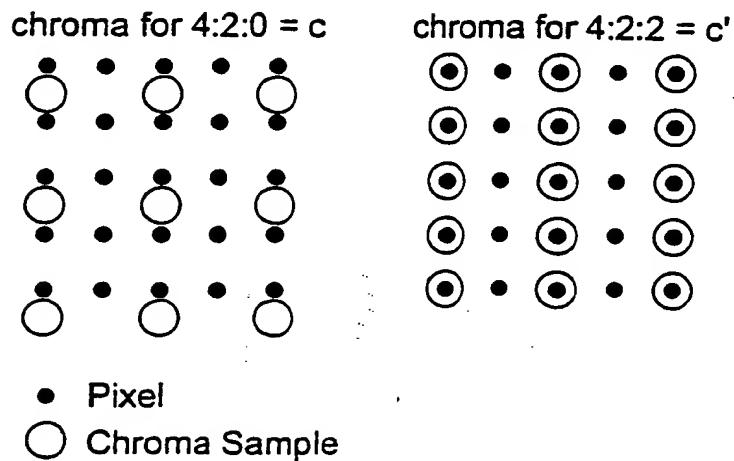


Fig. 14B
(YCbCr 4:2:2 to YCbCr 4:4:4 up-sampling)

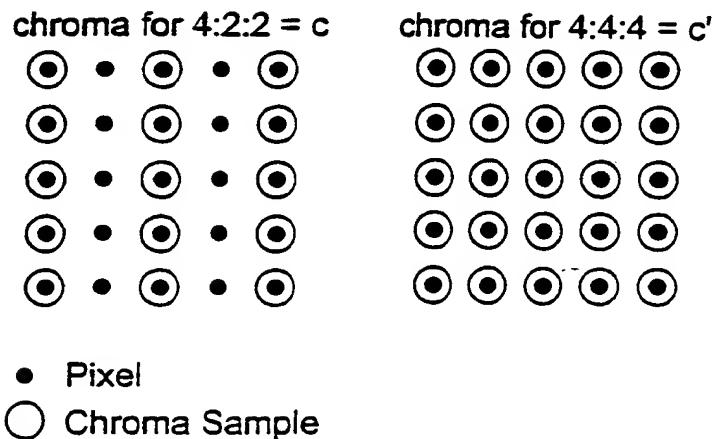
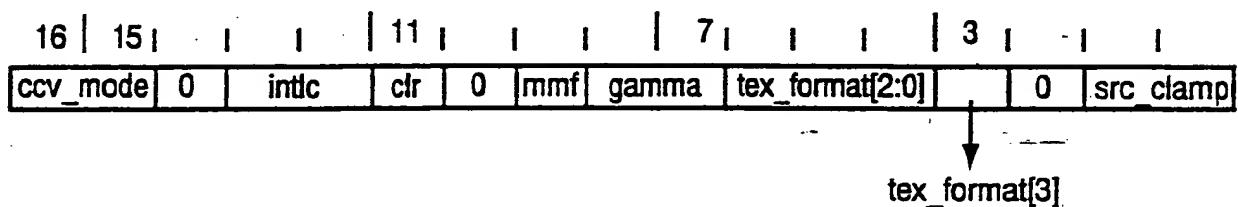


Fig. 15
(Control Register)



Fig. 16
(Texture Copy Command)

0x52:copy_cmd (texture) pixtypes allowed: rgb8, rgba6, rgb_aa,z,yuv8,yuv420:



0x52:copy_cmd (display) pixtypes allowed:
rgb8, rgba6, rgb_aa,yuv420:

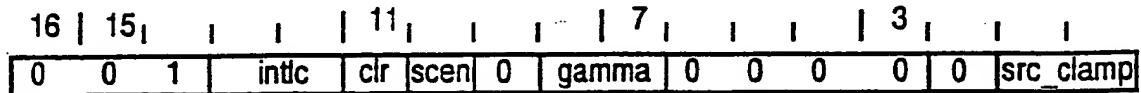


Fig. 17
(Display Copy Command)

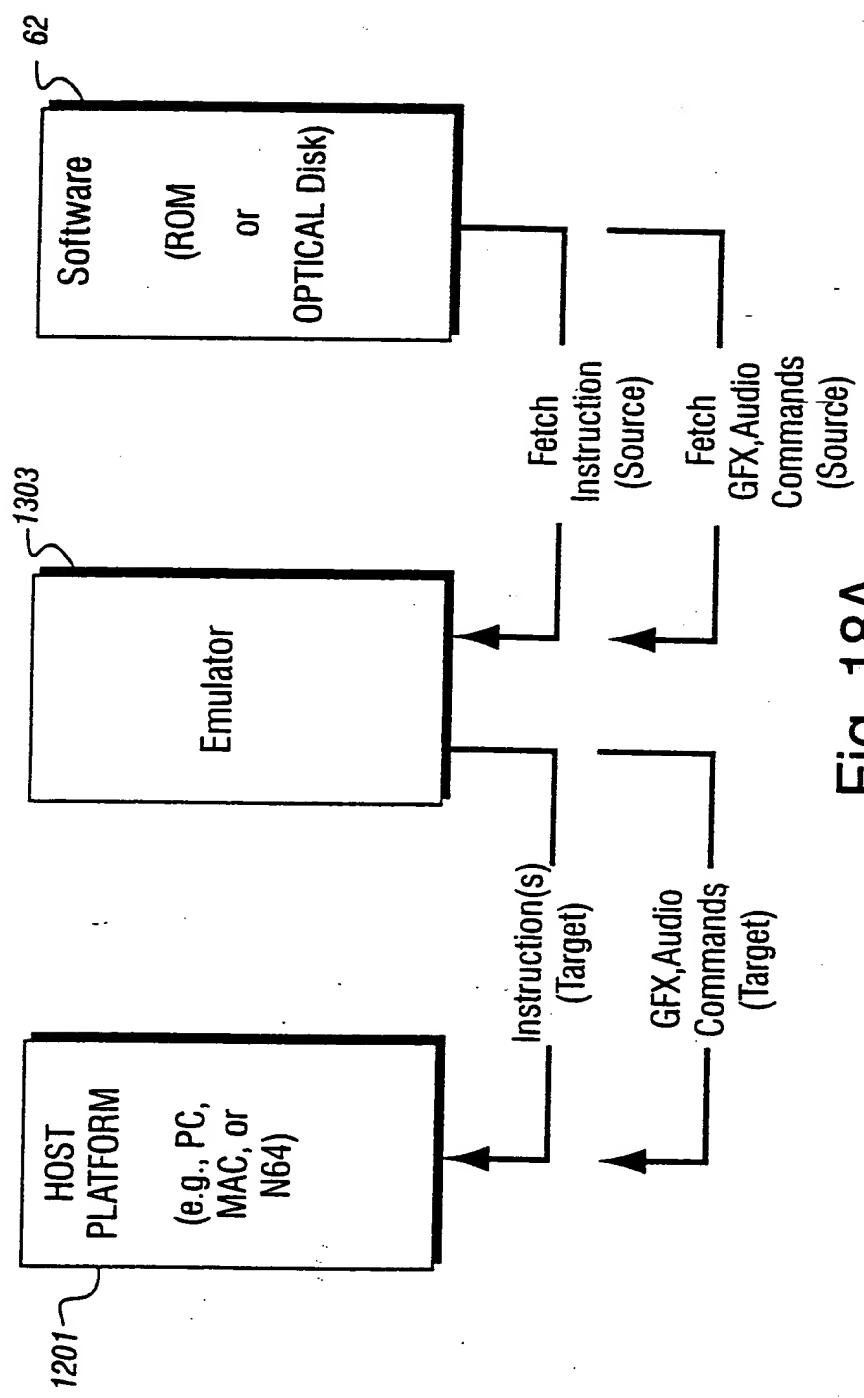


Fig. 18A

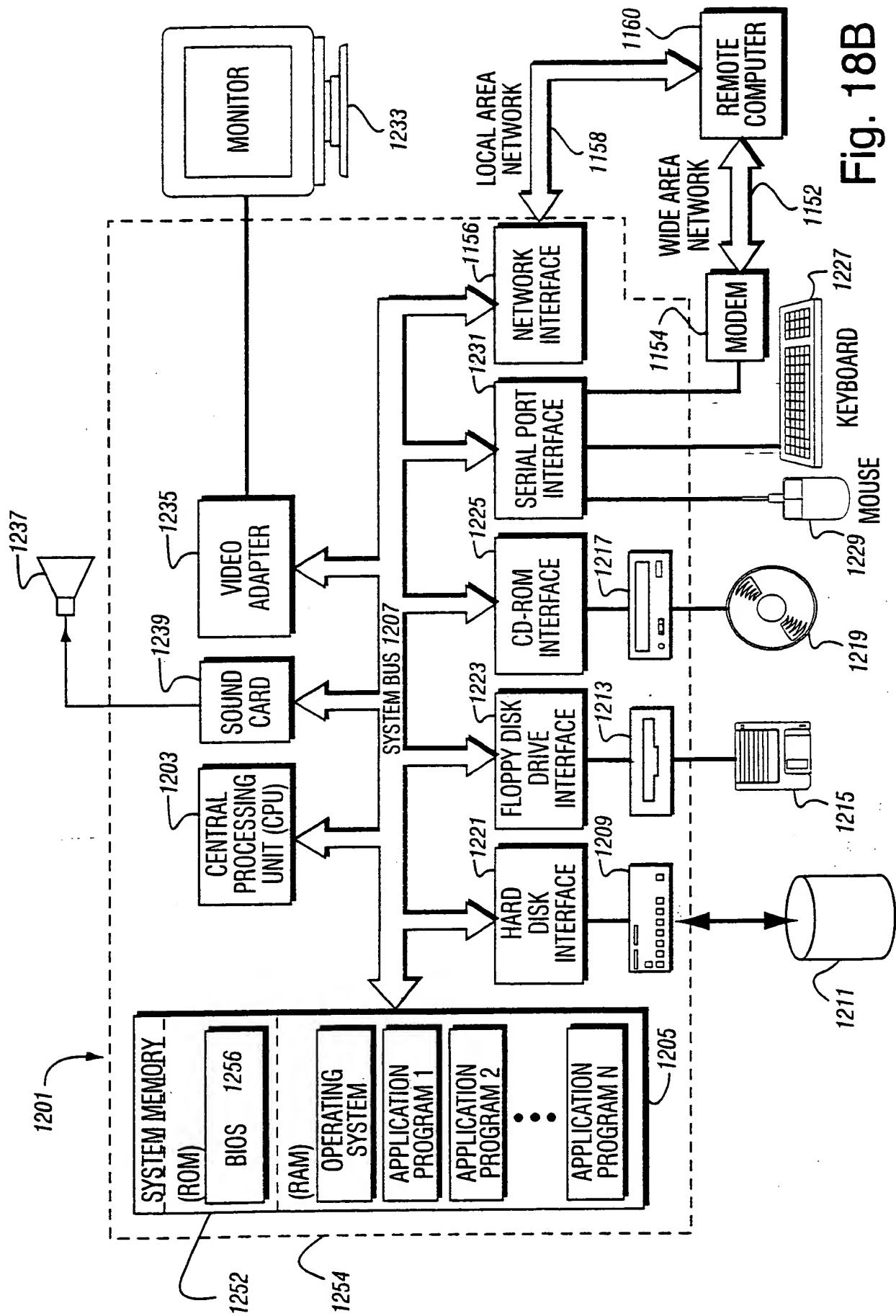


Fig. 18B